FALL 2024

Design and Development of a Blind Person Assistant

Team 6

Andy Lau
Emily Lai
Jeremy Mark Tubongbanua
Natasha Naorem

Advisor: Dr. Khalid Hafeez

Problem Statement

Major Concerns of Independent Navigation

Safety

Reliable obstacle detection

Efficiency

Reasonable travel time

Orientation

Self direction to detect surroundings and decide where to go

Existing Solutions

1. 3D Sound Systems
2. Ultrasonic Sensors
3. Smartphone Solutions
4. Map-based Systems

Survey Conclusion

Compared to outdoor solutions, there is a gap in the market for development of suitable indoor orientation/ navigation systems.

Problem Statement

Current indoor navigation systems do not have a focus on user-centered design

Current Solution

Obstacle Avoidance

Traditional Aids

- White Cane
- Guide Dog
- Sighted Guide

Orientation & Navigation

Mobile Applications

- Be My Eyes
- Google Maps
- Lazarillo
- Voice Vista

White Cane

Limitation:

Obstacle interaction is required to detect obstacles.

Guide Dog

https://www.akc.org/expert-advice/lifestyle/7-things-you-probably-didnt-know-about-guide-dogs/

Accessibility barrier: not all places welcome dogs or are accessible to dogs.

Ease of Use Barrier: 2 years to train and a formal application process

Lowest Borda Scoring Rule according to one study

Type of Assistance	Visually Impaired People		Sighted People	
	BSR	Ranking	BSR	Ranking
Sighted Guide	109	1	49	2
White Cane	76	2	63	1
Tactile Map	9	3	13	3
Guide Dog	6	4	13	3

Sighted Guide

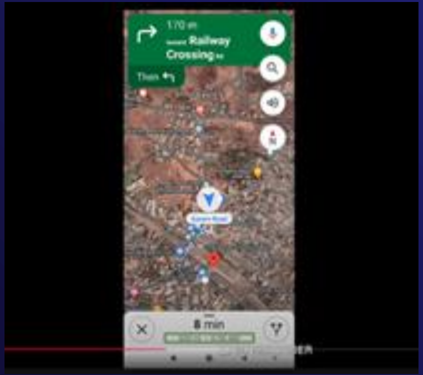
Limitation: Availability

Mobile Apps

Limitations

- Hindrance to use
- Few apps for indoor environments
- Not reliable at object detection

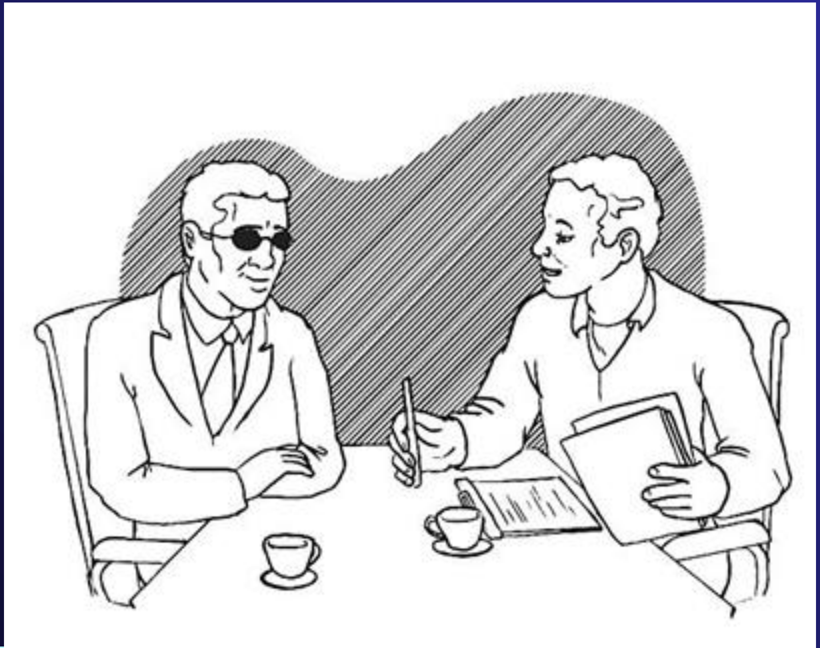
Google Maps



“Google Maps” by Google

- “Detailed voice guidance” → already includes solutions for the visually impaired
- Gives directions out loud
- Dynamically changes to known obstacles; such as construction
- Only good for broad Safety and Orientation

Stakeholder Discussion: Low Vision Student on Campus



Limitation: Disruptive Navigation

Simultaneous use of a white cane and mobile phone

Project Objectives

Improved Obstacle Avoidance

-> Increase detection range
-> Detect overhanging obstacles
-> Support the white cane

Non-Disruptive Navigation

-> Reduce reliance on simultaneous use of navigation tools
-> Improve navigation efficiency

Solution Design

Summary of our brainstorming iterations

Brainstorming Process

Individual Brainstorming

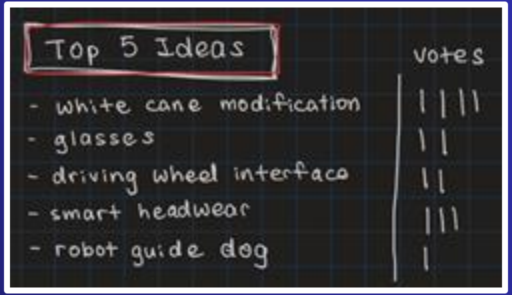
- Each member brainstormed at least 5 possible solutions

Group Brainstorming

- Each member presented their top 3 solutions to the group
- Solutions were workshopped: doubts clarified, similar ideas combined

Combined Voting

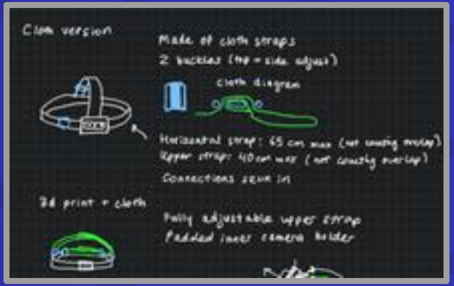
- Each member had 3 votes that they could distribute among project ideas
- Final vote results: Smart Headwear, Modified White Cane

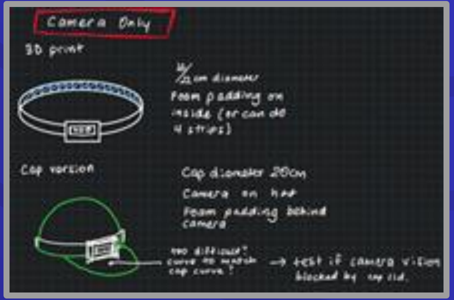


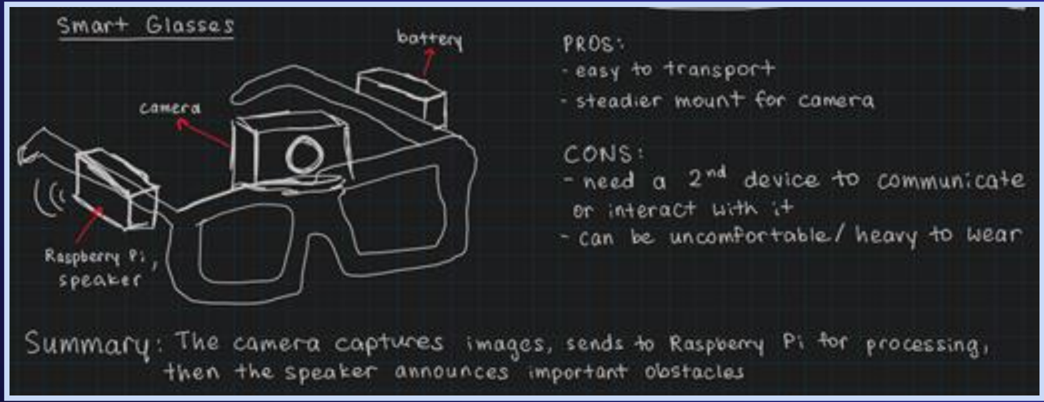
Brainstorming Session 1

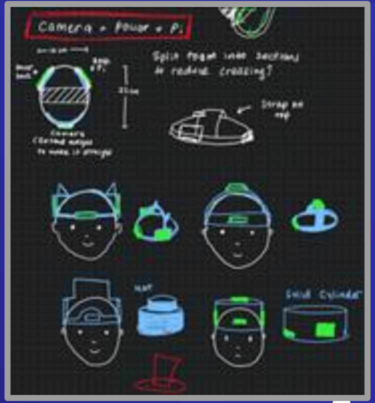
Smart Headwear

- Main considerations:
  - Wearability and comfort: weight, texture
  - Interaction methods: speech, buttons, etc.
  - Which elements should be included on the headwear (Raspberry Pi, speaker, battery)
- Inspiration: Meta-Ray Ban Smart Glasses





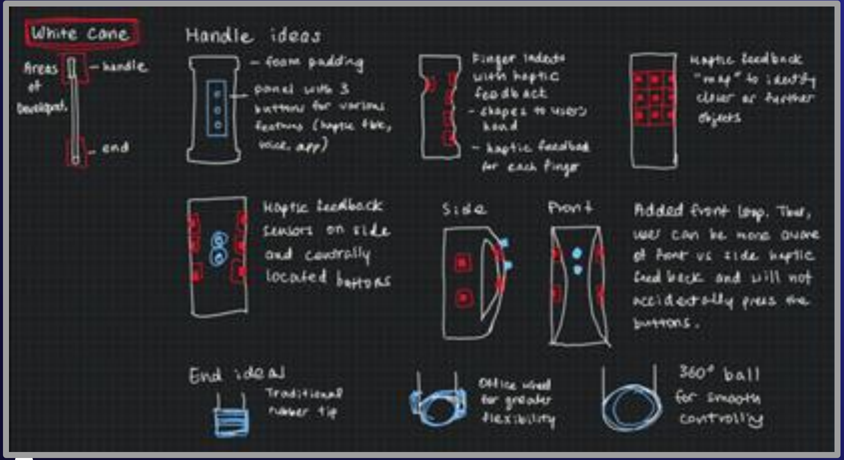


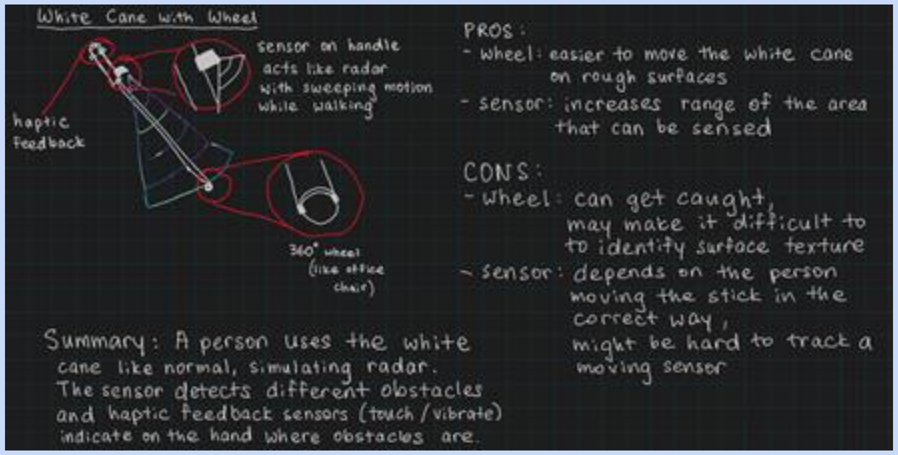


Brainstorming Session 2

White Cane Modifications

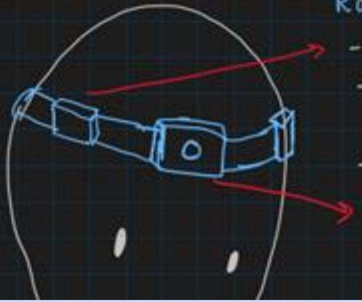
- Main considerations:
  - Keeping the traditional white cane structure
  - Keeping the traditional white cane movement
  - User interactions: buttons, haptic feedback
  - Placement of various elements





Prototype

Combination:
White Cane + Headset + App



Raspberry Pi + Speaker + Mic
- Pi connects to cellular hotspot
- Connects camera to speaker, GPS module, microphone
  ↳ updates every 3 mins
- 2 speakers to Bluetooth on phone

Camera
- Processes live video

Mobile App

- Settings
- View last location by connecting to cloud server

Cloud

- Text to speech processing
- Server that holds history (location of user)

Solution Architecture

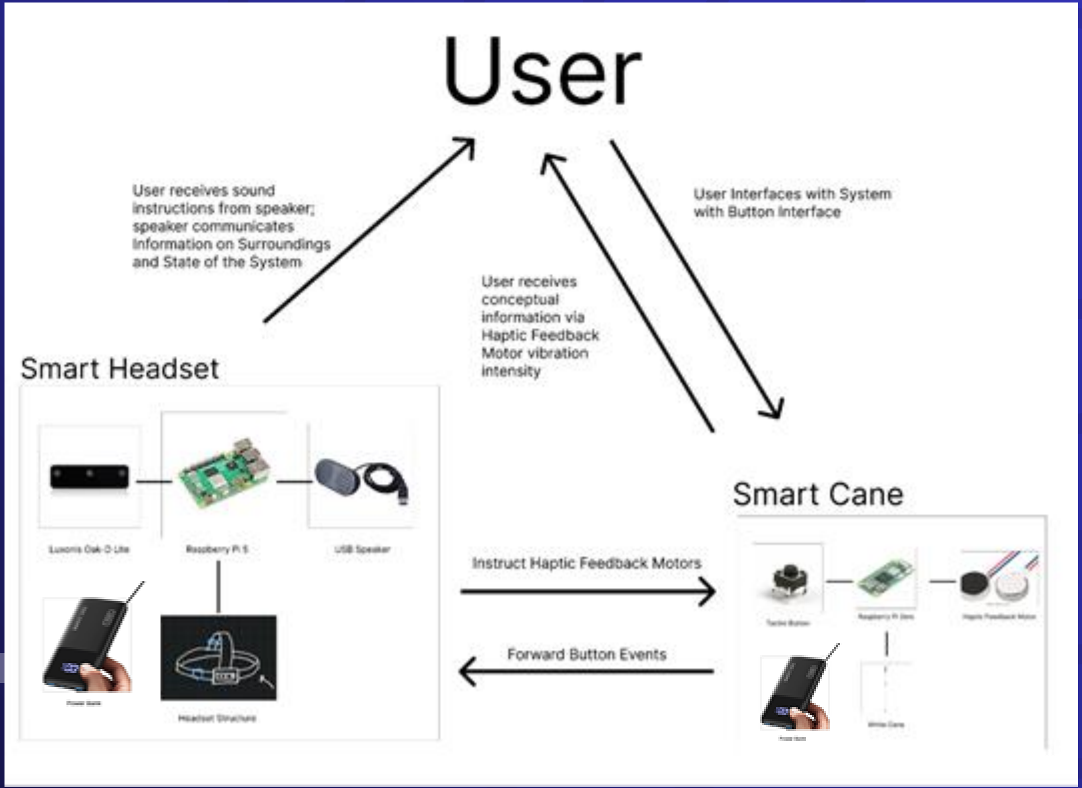
The architecture of our solution

Overall Architecture

User interfaces with system through smart cane and receives feedback from speaker and haptic feedback

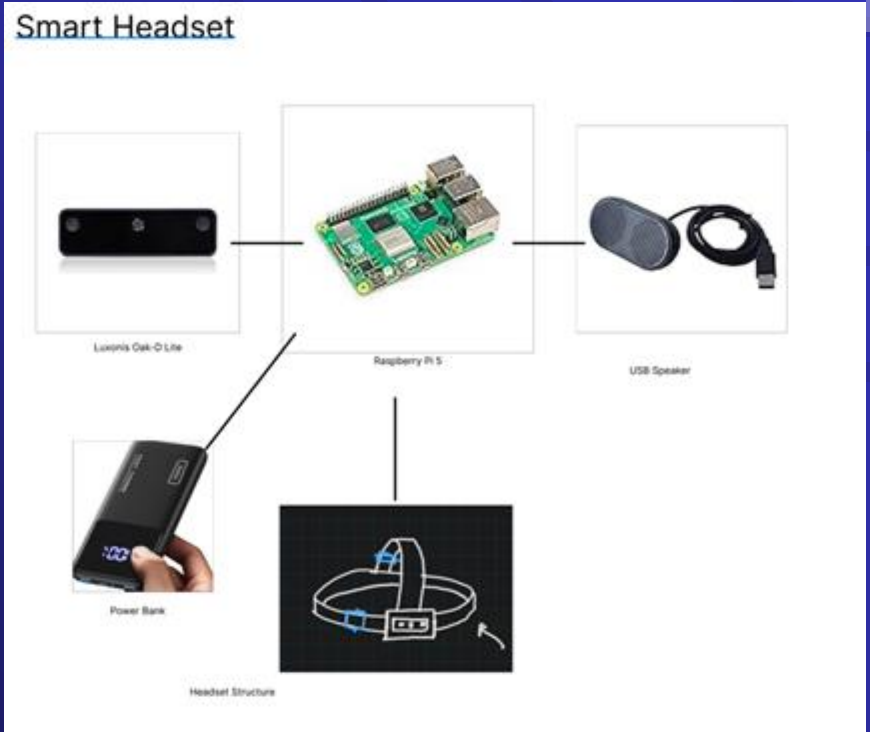
Two components connected with bluetooth:

- Smart Headset
- Smart Cane

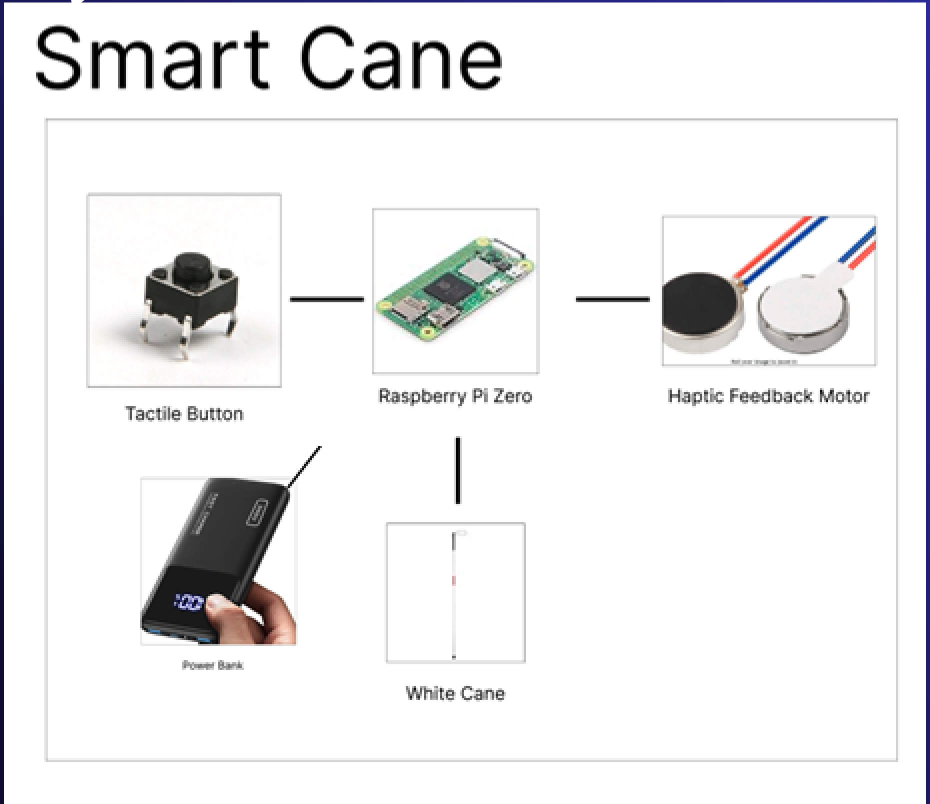


Headset Architecture

- Raspberry Pi powered by power bank
- Luxonis Oak-D Lite Depth AI Camera transmits object depth data and object detection and classification data to Raspberry Pi.
    - A model is trained specifically for our use case- to detect certain indoor objects
- Raspberry Pi controls speaker via USB



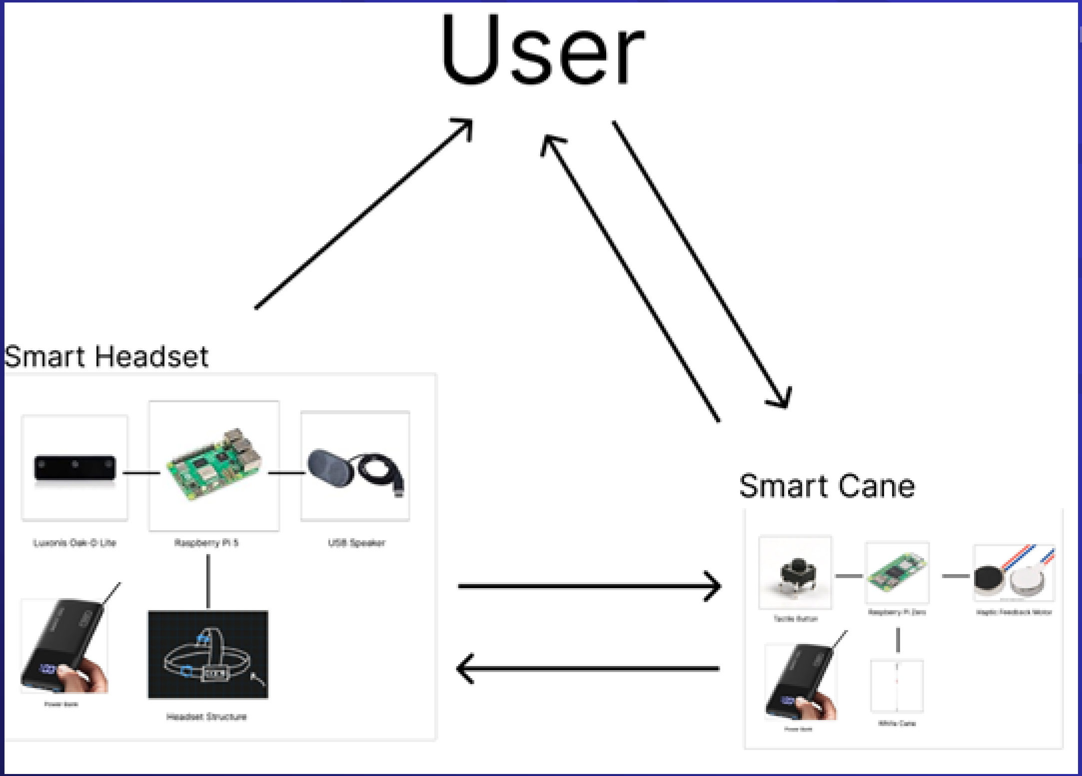
White Cane Architecture



- Raspberry Pi Zero acts as a secondary to the Raspberry Pi 5 connected to the camera and USB speaker. It is “dumb” in that it will only forward button events and receive instructions
- Haptic Feedback Motors - DC 3.3V motors
- Tactile Buttons - simple electrical switches
- Power Bank - for powering the system

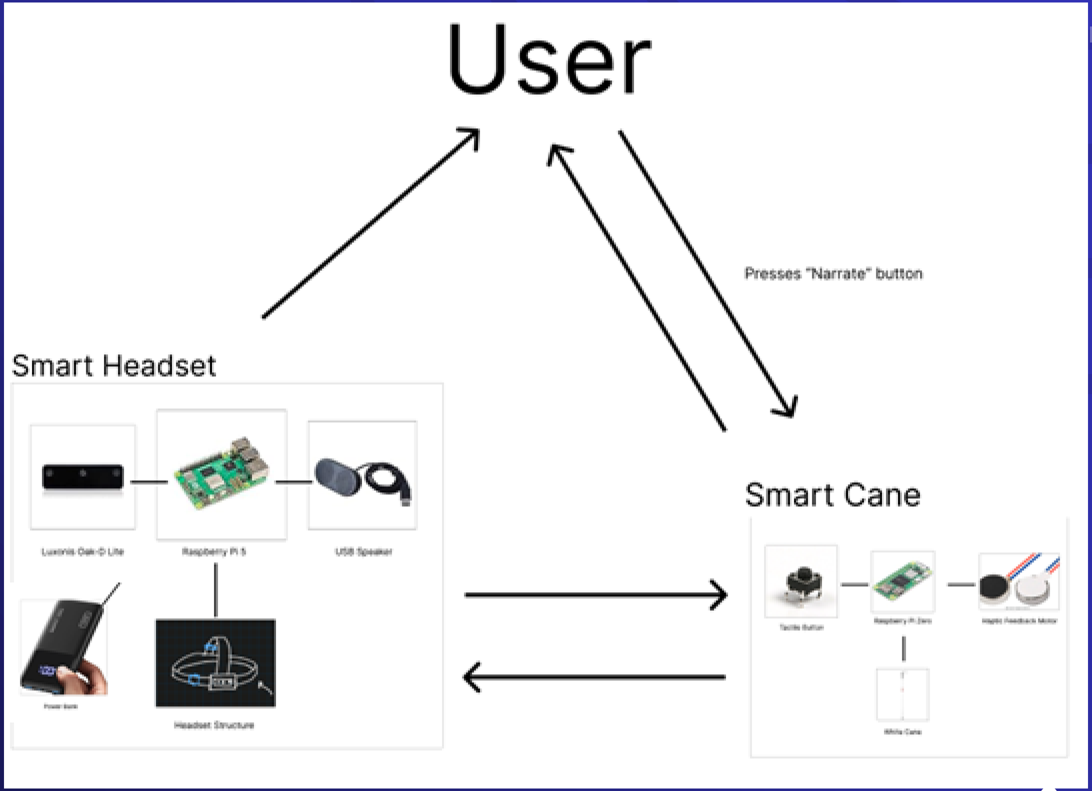
What a Hero Use Case Looks Like

The camera detects an object



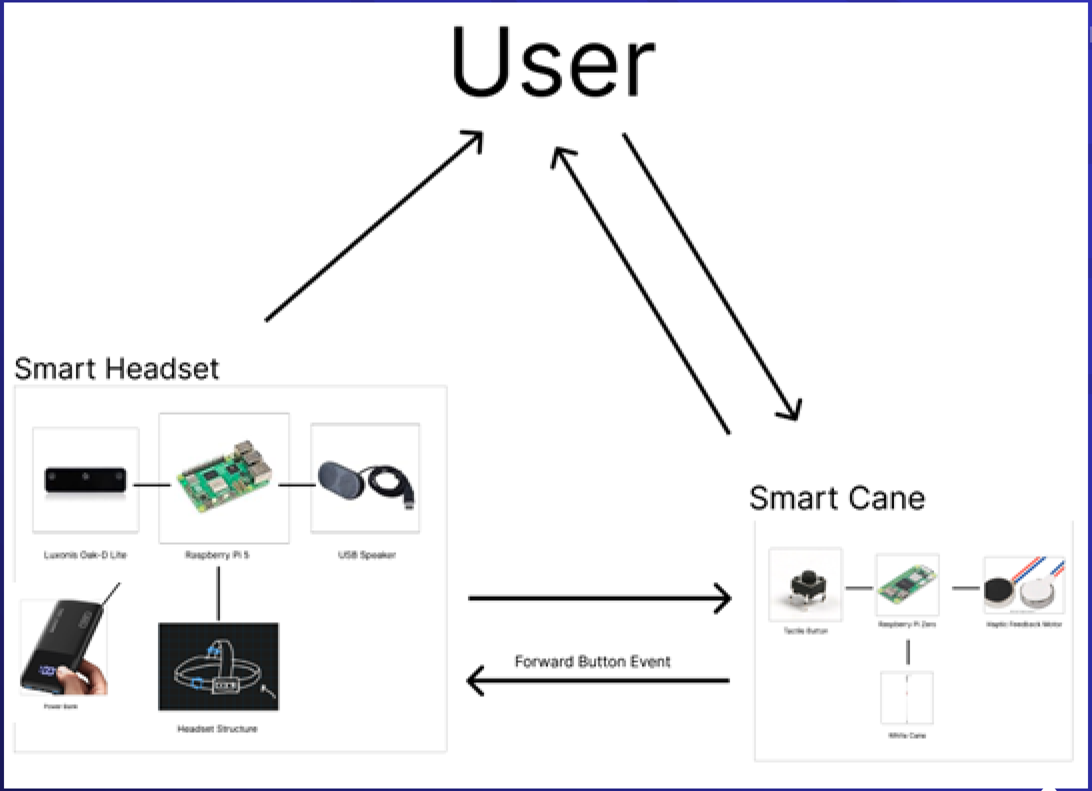
What a Hero Use Case Looks Like

The user presses the “narrate” button



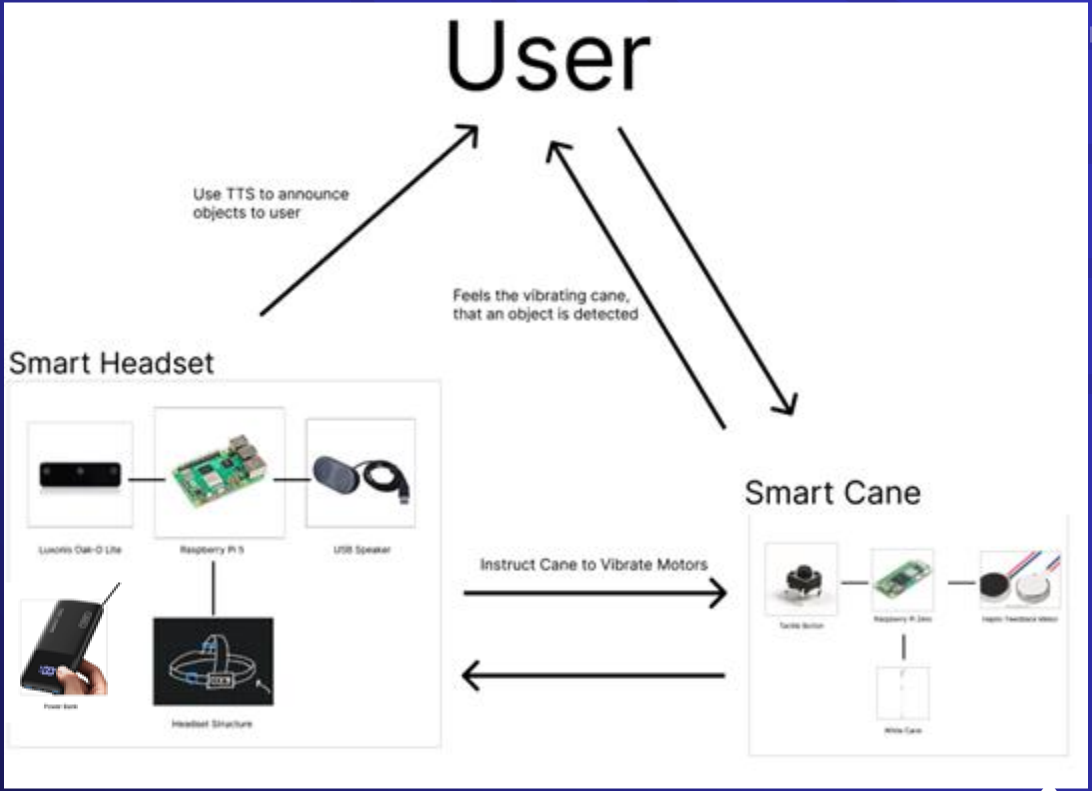
What a Hero Use Case Looks Like

The button event is forwarded to the master pi



What a Hero Use Case Looks Like

The Master Pi (headset) will use TTS (text-to-speech) to announce the objects to the user through the USB speaker and instruct the cane to vibrate.

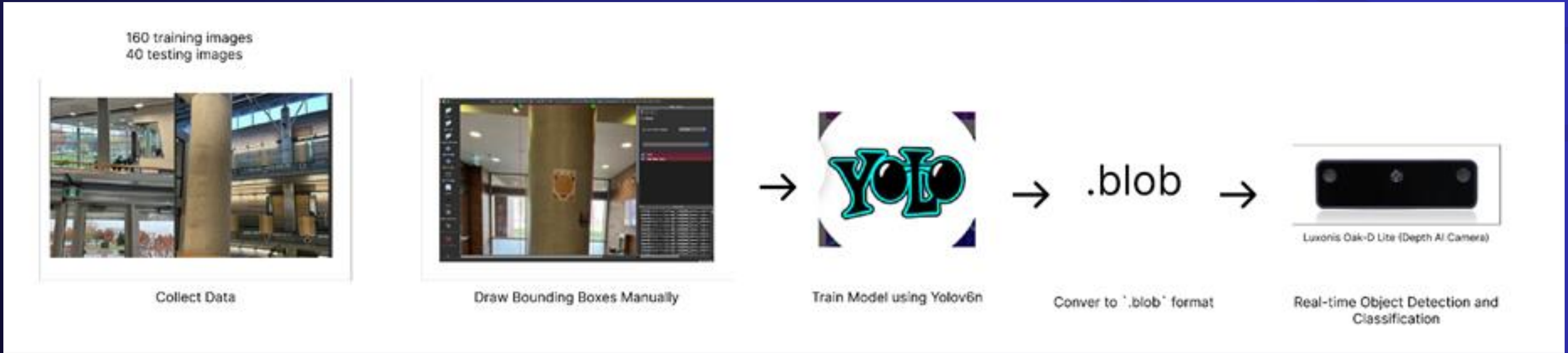


Training the Model

First we collected data

Training the Model

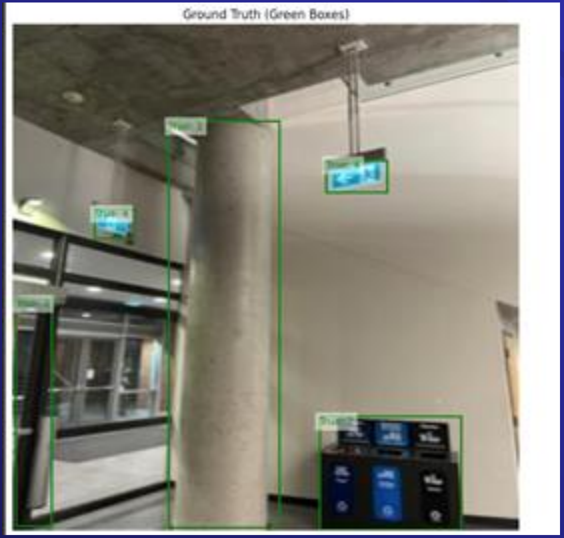
Model training process for object detection and classification



Unsuccessful Model

Trained with parameters:

- 10 epochs
- YOLOv6n base model
- 40 training images



Successful Model

Trained with parameters:

- 50 epochs
- YOLOv6n base model
- 160 training images

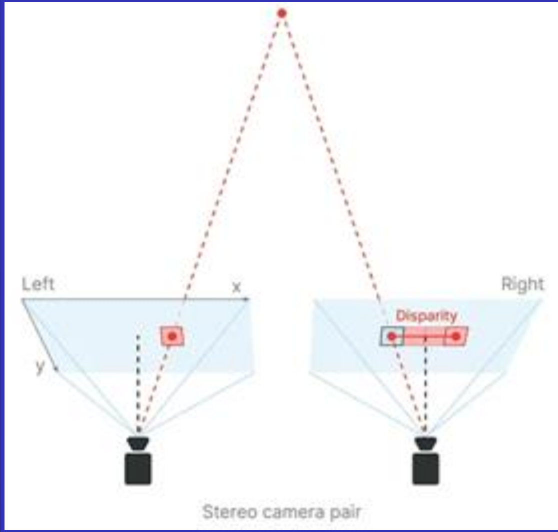
Depth Perception

Luxonis Oak-D Lite uses two monochrome cameras spaced 7.5cm apart.

Mimics human binocular vision to estimate depth.

VPU generates a depth map.

Oak-D Lite comes with limitation of depth perception range of [40cm, 8m]



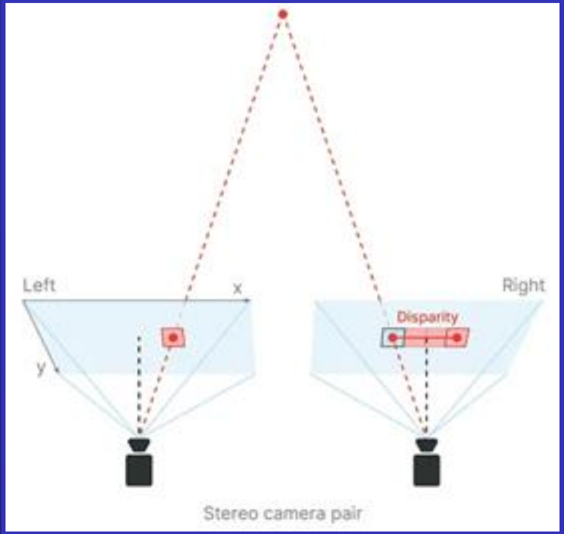
Depth Perception

Examples for calculating the depth value, using the OAK-D (7.5cm baseline):

```python
# For OAK-D @ 400P mono cameras and disparity of eg. 50 pixels
depth = 441.25 * 7.5 / 50 = 66.19 # cm

# For OAK-D @ 800P mono cameras and disparity of eg. 10 pixels
depth = 882.5 * 7.5 / 10 = 661.88 # cm
```

Note the value of disparity depth data is stored in `uint16`, where 0 is a special value, meaning that distance is unknown.



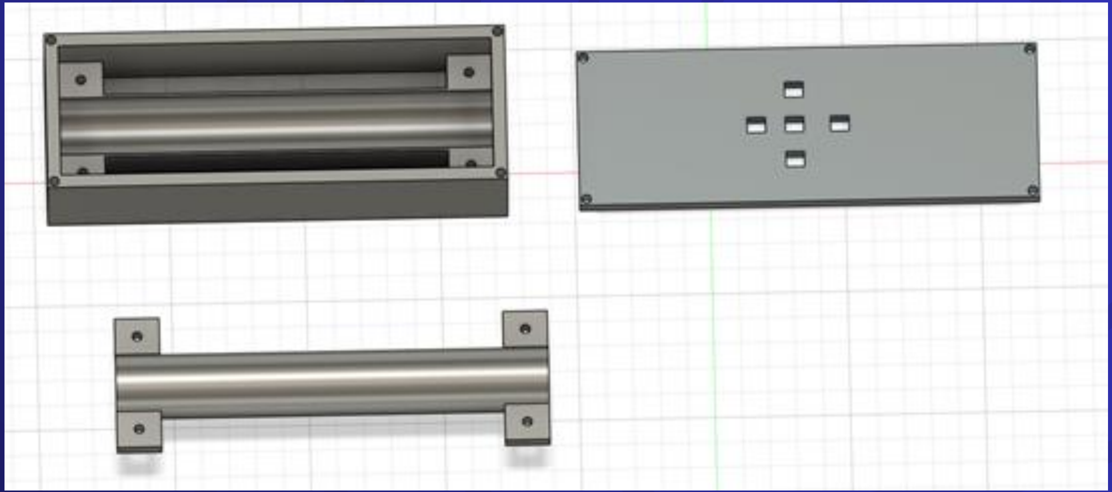
Theoretical calculations

- 441.25 → focal length in pixels (comes from calibration)
- 7.5 → base line in centimeters
- 50 → disparity in centimeters

Demo

UA Walkthrough

CAD Prototype of Smart Cane Housing

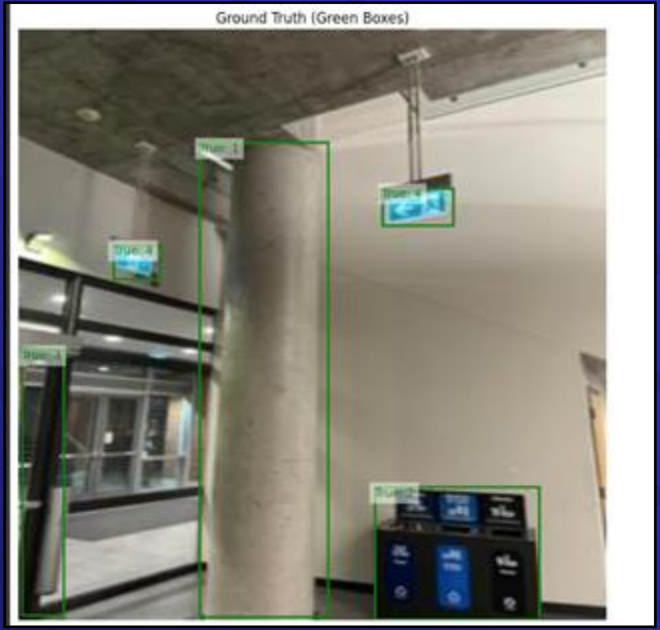


Test Results

Evaluation results and analysis

Test 1:
Spot Doors, Columns, Signs, and Trash Cans

- Camera spotting Doors, Columns, and Trash Cans should create a bounding box around the object



Test 2:
Estimate Distance to Obstacle

- Camera can tell the distance between it and objects it has been taught to spot

Thank You

Any Questions?